Automatic Generation of Gaze and Gestures for Dialogues between Embodied Conversational Agents: System Description and Study on Gaze Behavior

Werner Breitfuss[1] and Helmut Prendinger[2] and Mitsuru Ishizuka[1]

Abstract. In this paper we introduce a system that automatically adds different types of non-verbal behavior to a given dialogue script between two virtual embodied agents. It allows us to transform a dialogue in text format into an agent behavior script enriched by eye gaze and conversational gesture behavior. The agents' gaze behavior is informed by theories of human face-to-face gaze behavior. Gestures are generated based on the analysis of linguistic and contextual information of the input text. The resulting annotated dialogue script is then transformed into the Multimodal Presentation Markup Language for 3D agents (MPML3D), which controls the multi-modal behavior of animated life-like agents, including facial and body animation and synthetic speech. Using our system makes it very easy to add appropriate non-verbal behavior to a given dialogue text, a task that would otherwise be very cumbersome and time consuming. In order to test the quality of gaze generation, we conducted an empirical study. The results showed that by using our system, the naturalness of the agents' behavior was not increased when compared to randomly selected gaze behavior, but the quality of the communication between the two agents was perceived as significantly enhanced.

1 INTRODUCTION

Combining synthetic speech and human-like conversational behavior like gaze and gestures for virtual characters is a challenging and tedious task for human animators. As virtual characters are used in more and more applications, such as computer games, online chats or virtual worlds like Second Life, the need for automatic behavior generation becomes more pressing. Thus, there have been some attempts to generate non-verbal behavior for embodied agents automatically. Systems like the Behavior Expression Animation Toolkit (BEAT) allow one to generate a behavior script for agents by just inputting text [3]. The drawback of most current systems and tools, however, is that they consider only one agent, or only suggest behaviors, such that the animator still has to select appropriate ones by him- or herself. The aim of our work is to generate all non-verbal behavior automatically for conversing agents, so that someone writing a script to be performed by two agents can focus on creating the textual dialogue script and just feed it into the system. A salient feature of our system is that we generate the behavior not only for the speaker agent but also for the listener agent that might use backchannel behavior in response to the speaker agent. Employing two presenter agents holding a dialogue is advantageous, since watching (or interacting with) a single agent can easily become boring and it also puts "stress" on users, as they are the only audience. Furthermore, two agents support richer types of interactions and "social relationships" between the interlocutors. Also TV-commercials, games, or news use two presenters, because of the increased interaction possibilities and entertainment value.

In this paper, however, we confine discussion to the case where one user just watches the performance (dialogue) of two virtual agents, and does not interact with them. To assess the quality of our system we conducted an experiment. Twenty participants watched a presentation generated by our system. We randomly assigned them either to a version where the gaze behavior of the agents was informed by our gaze generator or to another version where the gaze was generated randomly. We speculated that the first (informed) version would increase the naturalness of the conversational behavior of the virtual characters and the quality of the communication between them. By "quality of the communication" we mean that the listener is paying attention to the speaker and the speaker addresses the listener in appropriate moments. In the study both versions used the same gestures, since we wanted to investigate the gaze behavior only. The dialogues where provided by a system developed at the Open University by Sandra Williams [20]. It generates a dialogue based on the medical history of a patient. While this system is designed to create shorter dialogues, for our purpose we used its original longer (unmodified) versions. The longer versions are sometimes repetitive, since patients in this database tend to have the some examinations over and over again.

The paper is organized as follows. In Section 2 we discuss related work. Section 3 describes our system and the way gaze behavior and other non-verbal behavior is generated by means of a "walk through" example. In Section 4 we describe our empirical study on gaze generation. The results are presented and discussed in the Section 5 and Section 6. Section 7 gives a short future outlook and concludes the paper.

2 RELATED RESEARCHES

Existing character agent systems already support the automated generation of some behaviors, such as automatic lip-synchronization. The next step is to automatically generate agents' conversational behavior from text. In this section, we report on some previous attempts, which combine various disciplines like computer animation, psychology, and linguistics.

2.1 Single Agent Systems

The BEAT system [3] generates synthesized speech and synchronized non-verbal behavior for a single animated agent. It uses plain text as input, which is then transformed into animated behavior. First, text is annotated with contextual and linguistic information, based on which different (possibly conflicting) gestures are suggested. Next, the suggested behaviors are processed in a 'filtering' module that eliminates gestures that are incompatible. In the final step, a set of animations is produced that can be executed, after necessary adoptions, by an animation system. The BEAT system can handle any kind of text and generate a run-able agent script automatically. The system uses a generic knowledgebase where information about certain objects and actions is stored, and the selected gestures are specified in a compositional notation defining arm trajectories and hand shapes independently, which allows the animator to add new gestures easily, or adjust existing ones.

The PPP Persona [1] is a life-like interface agent that presents multimedia material to a user. The behavior of the agent during the presentation is controlled partly by a script, written by the author of the presentation and partly by the agent's self-behavior. Behavior in the case of this agent is mostly acts such as pointing, speaking and expressing emotions and the automatically generated self-behavior which includes (1) idle-time actions to increase the personas life-like qualities, for example breathing or tipping a foot, (2) reactive behavior letting the agent react to external events like user reactions immediately, and (3) so-called navigation acts which display the movement of the agent across the screen, like jumping or walking. To generate this kind of behaviors a declarative specification language was used.

[13] describes a system that converts Japanese text into an animated agent that synchronously gestures and speaks. For assigning an appropriate gesture to some phrase the authors employed communicative dynamism (CD) as introduced by McNeill [12] and results from an empirical study that identified lexical and syntactic information and their correlation with gesture occurrence. For every "bunsetsu", the Japanese equivalent for a phrase in English, the system adds a gesture at a certain possibility, which is derived from the results of the study and the CD value. Similar to our system the specific gestures are defined in a library and if no specific gesture can be found for the bunsetsu, a beat is added as default gesture.

2.2 Multi Agent Systems

Another system is the eShowroom demonstrator[10], which was developed as a part of the NECA Project. The application automatically generates dialogues in a car-sales setting between an agent who acts as a seller and a second agent acting as buyer. The user has the possibility to choose certain parameters like topic, the personality and the mood of the virtual characters, which control the automatically generated dialogues. Also the gestures and behavior of the two screen characters would be generated by the NECA eShowroom demonstrator. It has three types of gestures: (1) turn taking signals like looking to the other interlocutor at the end of the turn, (2) discourse functional signals, which are gestures that depend on the type of the utterance (type refers to dialogue acts like inform or request), (3) feedback gestures are also generated to signal that the listener is paying attention to the speaker.

A different approach is suggested in [8]. This system supports the author in writing agent scripts by automatically generating gestures based on predefined rules, and using machine learning to create more rules from the set of predefined rules. It was used in the COHIBIT system, where the author first has to provide a script containing the actions for two virtual characters. In the next step the author writes simple gesture rules using his or her expert knowledge. Using this corpus of annotated actions the system can learn new rules. In the third step the system suggests the most appropriate gestures to the author, which are, after resolving conflicts and filtering, added to the already existing ones. Finally it produces a script with the gestural behavior of both virtual characters. Similar to our work, two agents are used, but since we want to reduce the workload to the minimum, our system does not require any input from the author except the dialogue to be presented by our characters.

2.3 Related work on eye gaze and gestures

[6] investigated the many different functions of gaze in conversation and its importance for the design of believable virtual characters. The gaze behavior of our agents is informed by empirically founded gaze models [7,15,19]. [7] analyzed gaze behavior based on two-person dialogs and found that gaze is used to regulate the exchange between the speaker and listener. In that work, different gaze patterns like the q-gaze (the speaker is looking at the person he/she is interacting with), and a-gaze (p is not looking at the interlocutor) were defined. It was found that the speaker looks at the listener while speaking fluently, but looks away when starting to speak or during hesitation (influent speech). In this way, speakers can keep the listeners attention or, by looking away, gain time to think about what to say next. Another finding is that mutual gaze can regulate the level of emotionality between interlocutors. The experiment described in [19] evaluates gaze behavior in multiparty environments, where four-person groups discussed current-affair topics in face-to-face meetings. Their results show that on average, interlocutors look about seven times more often at the speaker they listen to, than at others, and speakers looked about three times more at the addressed listener than at non-addressed listeners. Furthermore, the total amount of time spent gazing at each individual in a group of three is nearly 1.5 times higher than if visual attention of the speaking person were divided by three. These results are very relevant for our gaze algorithm since they give us the basis for a 'two agents' situation. And they also provide the needed information for our gaze generation rules. The work in [15] developed a model of attention and interest based on gaze behavior. An embodied conversational agent may start, maintain, and end a conversation dependent on its perception of the interests of the other agents.

Other related research was done is [5], which introduces a behavior synthesis technique for conversational agents in order to generate expressive gestures, including a method to individualize the variability of movements using different

dimensions of expression. The work described in [9] presents a gesture animation system that uses results from neurophysiologic research and generates iconic gestures from object descriptions.

3 GESTURE GENERATION SYSTEM

Our system consists of three different modules:

- Language Tagging module,
- Non-Verbal Behavior Generation module,
- Transformation to simple script or MPML3D module.

The Multimodal Presentation Markup Language is used to control the behavior of our 3D agents [14]. We choose a modular pipelined architecture to support future extensions. The code of the system is written in Java, and the XML format is used to represent and exchange data between modules.

The Language Tagging module takes the input dialogue text and uses the language module from the BEAT toolkit [3] to annotate linguistic and contextual information. Next, the Behavior Generation module adds non-verbal behavior like eye gaze and gestures to the annotated input sentence. In the final step, an agent script file is produced. In our implemented system, we can produce an MPML3D file but also a simpler script that can be used as an interface to other systems. The MPML3D player displays the embodied characters agents.

In our system, gaze patterns are generated for two different types of roles: (1) the speaker, i.e. the agent that is speaking and addresses the other agent, and (2) the listening agent. We can currently generate gaze behavior and gestures for these two roles, based on a given set of rules. Gaze directions have certain probabilities of occurrence, which we derived from existing gaze models [7,15,19]. In order to avoid conflicts between certain gaze behaviors, like looking in two different directions at the same time, we assigned priorities to them. Typically, more specific gaze behaviors (such as looking at speaker/listener) have higher priority than e.g. looking around. Moreover, we prioritize gazes that occur before starting the utterance, i.e., speakers typically look away before starting a long utterance (in order to concentrate on planning their dialogue contribution).

The rule in Figure 1 (adapted from [3]) shows one example of how the gaze behavior for the speaker is generated.

```
FOR each THEMA node in the tree
  IF at the beginning of the utterance
  Or 70% of the time
      Look away from listener
FOR each RHEMA node in the tree
  IF at the end of the utterance
  Or 73% of the time
      Look at listener
```

Figure 1. Gaze generation rule for the speaker

In addition to generating the gaze behaviour for the speaking agent we also have to consider the agent in the role of the listener. Since listeners typically look at speakers when they start an utterance (after taking floor) to demonstrate their attentiveness, we developed rules like the one in Figure 2.

```
FOR each THEMA node in the tree
  IF at the beginning of the utterance
  Or 80% of the time
      Look at speaker
FOR each RHEMA node in the tree
  IF at the end of the utterance
  Or 47% of the time
      Look at the speaker
```

Figure 2. Gaze generation rule for the listener

We also added gaze rules for certain gestures enacted by the speaker. For instance, pointing gestures have to be accompanied by the correct gazes. In our presentation scenarios we mostly use rectangular slides in the centre between the agents and smaller objects around them. As all of those objects have a definite position either left or right to the agent, we can exploit this knowledge to add the correct gaze direction to the agents' behavior when they talk about or point at the object. However, since defining the objects' position in the scener would increase the workload of the author, we also implemented the following straightforward principle. Every time a phrase such as "on my right side" or "to the left" occurs, we add a pointing gesture to the speaker's behavior tree. When the speaker's tree is completed, we recompile the listener's tree to adopt its gaze behaviour to the pointing gestures, and add the gestures to the correct side. The gestures of our agents are generated in similar manner, broadly following rules proposed in [3].

Let us now walk through one simple example utterance and see how our system works. As input we take the sentence: "This is just a small gaze example." [2] First, the input is sent to the Language Tagger module, which annotates the sentence with linguistic and contextual tags. The output of this process is shown in Figure 3. Here, "NEW" means that the word has not yet occurred in the conversation, and is thus a candidate for being accompanied by a "beat" gesture.

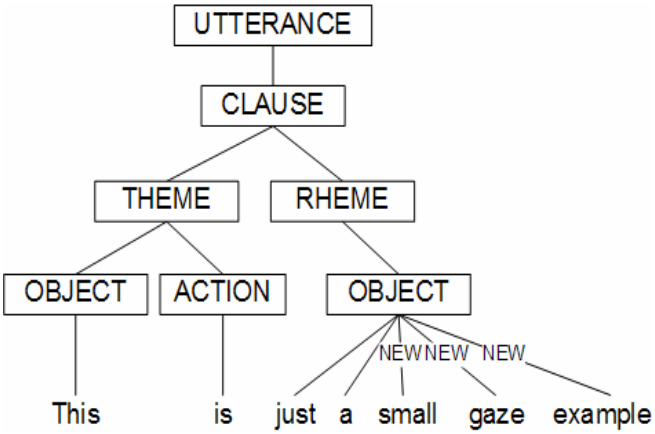

Figure 3. The output tree of the language module.

In the next step, we pass this newly constructed tree to our Behavior Generation module. It first generates a new tree with the gaze behavior and gestures (and speech parameters) for the speaker and a second tree for the listener. The tree for the listening character has the same structure as the speaker's tree, but contains the nodes for the non-verbal behavior that should be displayed by the listener agent.

Gestures are generated in two steps: first we add a beat every time some gesture is appropriate. After that the utterance is passed on to another layer that adds more specific gestures. To do this we provide a library, where we defined word bags associated with gestures. For instance, there is one word bag that contains the words "small, narrow, tiny" and the gesture for expressing something of little size. Hence, every time a word with the lemma of those words occurs in the sentence the beat gesture which has a lower priority is overwritten by the more specific gesture for small.

Figure 4 shows the speaker's tree, which was generated by our system for the sentence used in this short example. The root node of the tree is the utterance, and there is a speech pause between the theme and rheme of the sentence (see [3] for a discussion of speech parameters). The gaze behavior "Gaze away" and "Gaze at listener" is derived from the previously discussed rule (Figure 1). The gesture behavior is generated according to dedicated gesture generation rules of the Behavior Generation module. In our example, a beat gesture is selected to accompany the word "just", and an iconic gesture (for describing something small) is suggested to co-occur with the phrase "small gaze example".

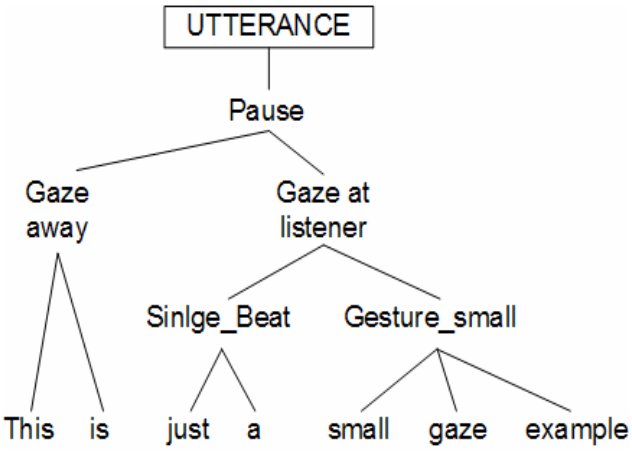

Figure 4. Tree for the speaker behaviour.

The behavior tree of the listener agent is generated similarly to that of the speaker agent (see Figure 5). It is based on the same tree that is output from the Language Tagging module of the speaker agent, but applies listener behavior generation rules instead of speaker rules. Again, we start with root node "UTTERANCE". During the speaker's speech pause, there is no behavior for the listener agent is defined. The listener's gaze behavior is added according to the rule in Figure 2, i.e. the listener is looking at the speaker when the utterance begins. Accordingly, our system creates the label "Gaze at speaker". Since the listener agent is paying attention to the speaker, it continues to look at the speaker also in the "rheme part" of the utterance.

Thereafter, appropriate gestures are suggested for the listener agent. Whereas no gesture is suggested for the phrase "just a", the phrase "small gaze example" is accompanied by head nods. In our system, a head nod is a basic gesture type for the listener. It is the gesture with the lowest priority and is used when no other, more specific gesture can be suggested. In the future, a dedicated "backchannel" knowledge base will be created to insert listener head nods in an informed, systematic manner.

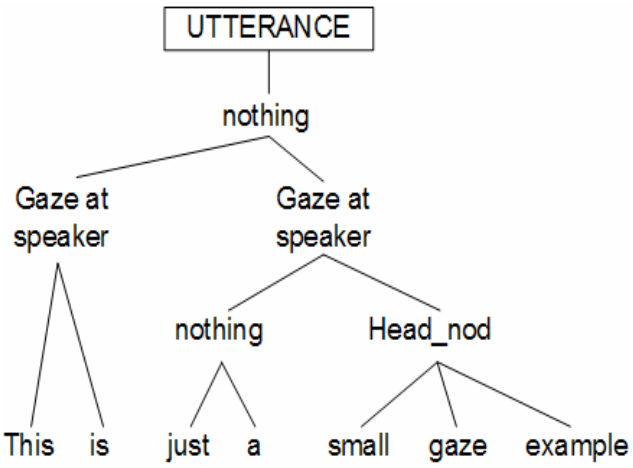

Figure 5. Tree for the listener behaviour.

After speaker and listener behavior trees are created, they are passed to the Transformation module, which compiles them into a synchronized MPML3D XML file or a simpler XML file.

Before generating the MPML Script we have to run the two trees through a small set of filters to handle any unexpected mistakes and to make sure no errors were forwarded to the script. We also use the filters to avoid minor technical problems like certain timing issues. Currently, the MPL3D Player cannot synchronize gestures that start at the beginning of a word and stop at the end of the same word.

This last module combines the speaker and listener tree by adding the actions of both agents for every utterance into one MPML3D structure called "task". The MPML Script contains parallel and synchronized actions which can be started and ended at the beginning, middle , or end of a certain word. First we add all the actions that should occur before the speaker starts to talk, mostly gaze behavior, like looking away from the speaker and idle gestures for the listener. The next action that is added is speaking itself. In the following step, we add the gaze behavior, which has to be aligned with the appropriate words. Gaze is implemented by having the head turn to a certain direction. There is a set of parameters that can be used, like the vertical ankle in which the head should be moved and the speed of the movement. As the last level we add the gesture for the speaking agent and the listening agent.

Figure 6 shows the MPML3D code, which our system generated for the sentence used in the example.

```
<Task>
  <Action>ken.turnHead(20,0.2,0.3,0.2)</Action>
 <Parallel>
  <Action name="kenspeak">
  ken.speak("This is just a small gaze example")
  </Action>
  <Action startOn="kenspeak[0].begin"
          stopOn="kenspeak[5].end">
  ken.turnHead(20,0.2,1,0.2)
  </Action>
  <Action startOn="kenspeak[6].begin"
          stopOn="kenspeak[14].end">
  ken.turnHead(0,0.2,5,0.2)</Action>
  <Action startOn="kenspeak[0].begin"
          stopOn="kenspeak[14].end">
  yuuki.turnHead(0,0.2,1,0.2)
  </Action>
  <Action startOn="kenspeak[6].begin">
  ken.gesture("beat_one")
  </Action>
  <Action startOn="kenspeak[17].begin">
  ken.gesture("showsmallvertical")
  </Action>
  <Action startOn="kenspeak[9].begin">
  yuuki.gesture("headnode")
  </Action>
 </Parallel>
</Task>
```

Figure 6. The MPML3D code for our example.

Our System can also produce a simpler script as output (see Figure 7). It contains only 3 entries: (1) the text of the utterance; (2) a mood, which is generated by using the [18] system, so that a virtual character that is able to display emotions, can use this information; (3) the gesture with the highest priority.

```
<utterance>
 <text> This is just a small gaze example</text>
 <mood>neutral</mood>
 <gesture>showsmallvertical</gesture>
</utterance>
```

Figure 7. Simple XML code for our example.

The simple script is intended to be used for other agent systems, which can only display one gesture per utterance or are limited with respect to gesture and speech synchronisation.
Figure 8 shows our agents performing the example sentence.

Figure 8. MPML3D Agents enacting the example sentence.

4 METHOD

4.1 Design

In the study, we compared two different versions of a presentation. In one version, gaze behavior was generated by our system (the informed version). In the control version, gaze was generated in a random manner (uninformed version). By "random" we mean that every time our system suggested a particular gaze behavior, a gaze direction was randomly chosen instead, which could be "look away" (to the left or to the right) or "look at the other agent".

The gestures used were the same in both versions, and consisted mostly of beats in case of speaking character, and head nodes in case of listening agent. We kept the set of the gestures used very limited, since as suggested in [4] too many gestures can distract the user and consequently have a negative effect on the perception of the overall presentation and gaze behavior.

We run the study primarily to investigate the effect of our new gaze module on two dimensions: (1) the naturalness of the presentation, and (2) the perceived quality of the conversational behavior between the two agents. The dialogues where generated by the [20] system.

4.2 Participants

Twenty people participated in the study, 18 males and 2 females, their age ranged from 22 to 35 years (mean age 28.3 years). Except for two external people, subjects were students or researchers from the National Institute of Informatics, Tokyo. Subjects received 1000 Yen for participating.

4.3 Materials

The raw dialogues for the presentation were provided by an automated dialogue generation system [20], and contain the conversation between Yuuki, a female senior nurse and Ken, a male junior nurse.
The dialogue contained 106 utterances, and the duration of the presentation was around 5 minutes. The topic of the dialogue was about the medical history of a fictional patent that has breast cancer.

The following is a typical paragraph of the presentation. We wish to note again that for the purpose of the experiment (investigating gaze), we used the long, unmodified dialogue output by the system. This output was not meant to be shown to subjects when investigating e.g. the effectiveness of the dialogue.

Yuuki: For May the 24th what does the medical record say?
Ken: On May the 24th she did a self examination.
Yuuki: What did she find?
Ken: A lump.
Yuuki: What does it say next?
Ken: On May the 19st she did another self examination.
Ken: And she still had a lump.
Yuuki: And then?
Ken: On June the 7th she did another self examination.

Ken: And she still had a lump.
Ken: From May the 20th to August the 5th she had a chemotherapy course.
Ken: What is a chemotherapy course ?
Yuuki: A chemotherapy course is a treatment with drugs.
Yuuki: Is that clear ?
Ken: Uhhuh.
Yuuki: What does it say next ?
Ken: On June the 24th she had another examination.
Ken: And she still had lymphadenopathy.

4.4 Apparatus

The experiment was run on a Dell workstation with a dual-core processor. The material was presented to the subjects using a UXGA (1600 × 1200 pixels) flat screen color monitor. The speech for the agents was generated by Loquendo ([11]), a commercial text-to-speech (TTS) engine. The agents controlled by our MPML3D Player ([14]).

For videotaping the participants we used a digital camera that was positioned behind subjects and a mirror, which was fixed on the right side of the monitor, so that we could capture the face and the shoulders of the subjects. Figure 9 depicts the setup of our study.

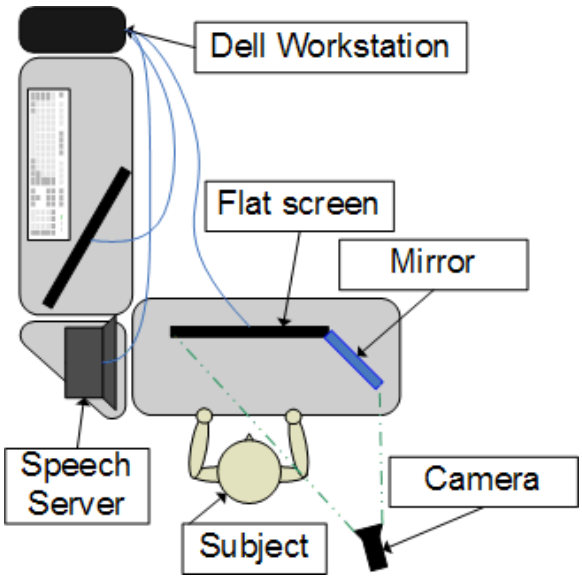

Figure 9. Experimental setup.

4.5 Procedure

Subjects entered the experiment room individually and received a written instruction about the procedure. The instruction given to the subjects was to watch the presentation as they would watch a presentation given by human presenters and they should keep an eye on the behavior of the agents.
While watching the dialogue between our two agents, the participants were videotaped for further analysis (Figure 10: screen with presentation to the left, participant to the right).

After watching the presentation, both groups of participants were asked to fill out a questionnaire with twelve questions.

1. The female agent (Yuuki) was friendly.
2. The male agent (Ken) was friendly.
3. The conversation between the two agents seemed very natural.
4. Sometimes I thought the agents react to each other in a strange way.
5. I felt that the two agents are a good team and communicate with each other well.
6. It seemed that the agents did NOT pay attention to each other.
7. I trusted the female agent (Yuuki).
8. I trusted the male agent (Ken).
9. I found the conversation easy to follow.
10. The conversation captured my attention.
11. I found that my attention wandered.
12. I found the conversation hard to understand.

The answers were based on a Likert scale, and ranged from 1 ("strongly agree") to 7 ("strongly disagree"). At the end of the questionnaire we also provided the possibility of free text entry, so that subjects could state their comments without restrictions.

Each session of the experiment lasted around 15 minutes per person, and was conducted in our multimedia room.

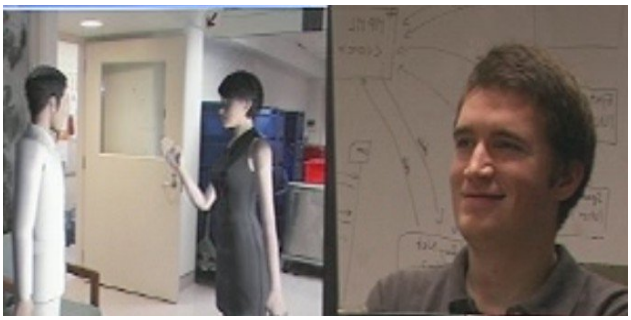

Figure 10. Screen and participant.

5 RESULTS

We performed a t-test[1] (two-tailed) to determine the statistical significance of the differences between the averages (significance level α set to .05).

The averages of the answers to the questions in the questionnaire can be found in Figure 11 where the x-axis gives the number of the question, and the y-axis shows the value for each question. Figure 12 shows the means and standard deviations of the questions Q1 to Q12, where the first row gives

[1] The *t*-test tells us how likely it is that the means of the two populations are equal based on actual distance between the means and the within group variability of the two groups. The magnitude of *t* increases as the distance between the means increases and the within-group variability decreases. As *t* increases, the probability of the means being equal, decreases.

the values, mean and deviation, of the uninformed version and the second row gives the values for the informed version.

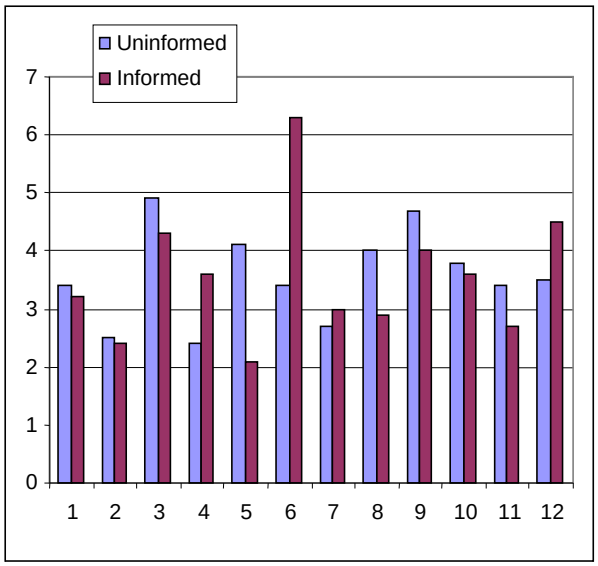

Figure 11. The means for the questions.

Question nr.:	Q1	Q2	Q3	Q4
mean and deviation uniformed version	3.4 ± 1.1	2.5 ± 0.7	4.9 ± 1.4	2.4 ± 1.1
mean and deviation informed version	3.2 ± 1.4	2.4 ± 0.8	4.3 ± 1.6	3.6 ± 1.3
Question nr.:	Q5	Q6	Q7	Q8
mean and deviation uniformed version	4.1 ± 1.4	3.4 ± 1.4	2.7 ± 1.3	4 ± 1.4
mean and deviation informed version	2.1 ± 0.7	6.3 ± 0.5	3 ± 1.3	2.9 ± 0.9
Question nr.:	Q9	Q10	Q11	Q12
mean and deviation uniformed version	4.7 ± 1.6	3.8 ± 1.4	3.4 ± 1.3	3.5 ± 1.4
mean and deviation informed version	4 ± 1.8	3.6 ± 1.8	2.7 ± 1.3	4.5 ± 1.6

Figure 12. Means and standard deviations.

We predicted that the gaze behavior generated by our system would generate a more natural dialogue and the agents would be perceived as communicating well which each other.
Regarding the first dimension (naturalness), we partly obtained significant results, while always showing the expected tendency in the answers (Questions 3 and 4). The results for the question concerning the naturalness of the agents' behavior, the results for Question 3 showed that the informed version only slightly improved the naturalness of the conversation ($p = .396$). The result for Question 4 though showed that the agents reacted significantly less strange (by contraposition, more natural) to each other in the informed version ($p < .041$).

The results for questions concerning the conversational behavior between the agents (quality of communication) are statistically significant. The results confirm the hypothesis that our system can significantly increase the level of perceived quality of conversational behaviour between the two interlocutors For Question 5, $p < .0017$, and for Question 6, $p < .0001$.

The questions regarding the friendliness of the agents (Questions 1 and 2), or about the trustworthiness of the agents (Questions 7 and 8), did not yield any important results. Note, however, that the results for Question 8 indicate that the male character was nearly significantly ($p = .053$) more trustworthy in the version informed than in the uninformed version.

6 DISCUSSION

The purpose of the experiment was to obtain empirical data on our newly implemented system, with a focus on the gaze behavior of the agents. The data from the questionnaires supports our expectations that the version with gaze behavior informed by our system would outperform the version with randomized gaze in terms of quality of conversational behavior between the two embodied virtual characters. In particular, the result for Question 6 provides strong evidence that the participants noticed that the agents pay more attention to each other in the informed version.

The poor results regarding the naturalness of the presented dialogues were somewhat surprising. The free-text comments we received from the participants (as part of the questionnaire) gave three different reasons why they rated the naturalness as rather poor. One issue was the beat gesture, which seemed to be irritating, and the hand movement was too fast and too wide. A second problem was the voice generation, which did not produce satisfying results for technical medical terms. (In fact, this problem could have been avoided if we had provided the correct pronunciation of rare technical terms to the TTS engine beforehand.). Third, some subjects criticized parts of the dialogue as unnatural. They noted that there are too many repetitions and some of the answers given by the junior nurse (Ken) were irritating. There is one particular part in the dialogue, where the senior nurse explains the function of auxiliary lymph nodes, and the junior nurse answers with a short "Cool". As the video analysis showed, most participants found this part rather humorous, but others stated in their comments, that it is strange to use the word "cool" in the context of cancer. The experiences with our study provide highly valuable insights for designing better studies with our non-verbal behavior generation system in the future.

7 CONCLUSIONS AND FUTURE WORK

There is ample evidence that agent-based multimodal presentations can entertain and engage the user, and are also an effective way to mediate information [17]. In this paper, we described our system that automatically generates gaze and gestures for two agents, in the roles of speaker and listener. It uses a dialogue script as its only input (from the content creator), and transforms it into a run-able multimodal presentation using two highly realistic 3D character agents.

In our future work, we plan to analyze the emotional content of text based on the work described in [18], and add emotional expressions to the agents' behavior in order to improve the naturalness of the performed dialogue. The emotion expressed in a sentence will also affect voice parameters, gaze, and gesture

behavior. Conversational behavior is also influenced by the social role (instructor-student, employer-employee, etc.), the cultural background, and the personality of the interlocutors. Another venue of research relates to including a model of the user as a listener, who might be addressed by the agents.

Our next step, however, will address more feasible issues. In addition to extending the set of behavior generation rules for the listener agent, we want to align the behavior of the agents with respect to a slide show and virtual objects in a 3D environment. Here, we have to analyze phrases like "if you look at the slide" and generate appropriate behavior for the speaker and listener agent. Among others, the selected gaze behavior has to be timed and directed to specific locations in the 3D environment. In this way, "joint attention" (gaze) behavior will be implemented.

For all of our ideas, the focus will remain on the exploration of ideas that ultimately lead to a minimal workload for content creators, while ensuring high-quality, professional output in the form of natural and enjoyable multimodal presentations.

8 REFERENCES

[1] André, E., Müller, J., and Rist, T.: The PPP Persona: A Multipurpose Animated Presentation Agent In: Catarci T., Costabile M.F., Levialdi S., and Santucci G., editors, Advanced Visual Interfaces, pages 245-247. ACM Press (1996)

[2] Breitfuss W., Prendinger H., Ishizuka, M.: Automated Generation of non-verbal behavior for virtual embodied characters. In: Proceedings of the 9th international conference on Multimodal interfaces, pages 199 -202 (2007)

[3] Cassell, J., Vilhjálmsson, H., and Bickmore, T.: BEAT: the Behavior Expression Animation Toolkit. In: Proceedings of SIGGRAPH 2001, pages 477-486 (2001)

[4] Craig, S., Gholson, B., and Driscoll, D. Animated pedagogical agents in multimedia educational environments: Effects of agent properties, picture features, and redundance. Journal of Educational Psychology, 94(2); pages 428 – 434, (2002)

[5] Hartmann, B., Mancini, M., Buisine, S., and Pelachaud, C.: Design and evaluation of expressive gesture synthesis for embodied conversational agents. In: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems. ACM Press (2005)

[6] Heylen, D.: Head gestures, gaze and the principles of conversational structure, In: International Journal of Humanoid Robotics Vol. 3 Nr. 3, pages 241-26 (2006)

[7] Kendon, A.: Some functions of gaze-direction in social interaction. In Acta Psychologica 26, pages 22-63, North-Holland Publishing Co. (1967)

[8] Kipp, M.: Creativity meets automation: Combining nonverbal action authoring with rules and machine learning, In: Proceedings of the 6th International Conference on Intelligent Virtual Agents (IVA 2006), Springer, pages 217-242 (2006)

[9] Kopp, S., Tepper, P., and Cassell, J.: Towards integrated microplanning of language and iconic gesture for multimodal output. In: Proceedings International Conference on Multimodal Interfaces 2004, ACM Press, pages 97-104 (2004)

[10] Krenn, B., Grice, M., Piwek, P., Schroeder, M., Klesen, M., Baumann, S., Pirker, H., van Deemter, K., and Gstrein, E.: Generation of multi-modal dialogue for net environments. In Proceedings of KONVENS-02, pages 91–98, (2002)

[11] Loquendo Vocal Technologies and Services, URL (2008) www.loquendo.com

[12] McNeill, D.: Hand and Mind: What Gestures Reveal about Thought. Chicago, IL/London, UK: The University of Chicago Press. (1992)

[13] Nakano, Y., Okamoto, M., Kawahara, D., Li, Q., and NishidaT.: Converting Text into Agent Animations: Assigning Gestures to Text, In: Journal of Humanoid Robotics Vol. 3 Nr. 3, pages 241-26 (2006)

[14] Nischt M., Prendinger, H., André, E., and Ishizuka M.: MPML3D: a reactive framework for the Multimodal Presentation Markup Language. In: Proceedings 6th International Conference on Intelligent Virtual Agents, Springer, pages 218-229 (2006)

[15] Peters, C., Pelachaud, C., Bevacqua, E., and Mancini, M.: A model of attention and interest using gaze behavior. In: Proceedings of 5th International Conference on Intelligent Virtual Agents 2005, pages 229-240 (2005)

[16]Prendinger, H. and Ishizuka, M., editors. Life-Like Characters. Tools, Affective Functions, and Applications. Cognitive Technologies. Springer, Berlin Heidelberg, (2004)

[17] Rist, T., André, E., Baldes, S., Gebhard, P., Klesen, M., Kipp M., Rist, P., and Schmitt, M.: A review of the development of embodied presentation agents and their application fields. In: Prendinger and Ishizuka [16], pages 377-404

[18] Shaikh, M., Prendinger, H. and Ishizuka, M.: A Cognitively Based Approach to Affect Sensing from Text. In: Proceedings 10th International Conference on Intelligent User Interfaces, ACM Press, pages 349-351 (2006) Computing Systems (CHI'03), pages 521–528, ACM Press (2003)

[19] Vertegaal, R., Weevers, I., Sohn, C., and Cheung, C.: Gaze-2: conveying eye contact in group video conferencing using eye-controlled camera direction. In: Proceedings of the SIGCHI Conference on Human factors in

[20] Williams, S., Piwek, P., and Power, R.: Generating monologue and dialogue to present personalised medical information to patients. In Proceedings of the 11th European Workshop on Natural Language Generation, pages. 167-170 (2007)